

PATENT
N0070US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
APPLICATION FOR UNITED STATES LETTERS PATENT

INVENTORS: THOMAS K. HUBER
WILLIAM GALE

TITLE: MULTI-DIMENSIONAL SPATIAL
INDEX FOR A GEOGRAPHIC
DATABASE

ATTORNEYS: Frank J. Kozak
Lawrence M. Kaplan
NAVIGATION TECHNOLOGIES
CORPORATION
Rosemont, Illinois 60018
(847) 795-7000 x7371

00907 42630460

1 MULTI-DIMENSIONAL SPATIAL
2 INDEX FOR A GEOGRAPHIC DATABASE

3 BACKGROUND OF THE INVENTION

4 The present invention relates to geographic databases and more particularly, the
5 present invention relates to an index that is useful for accessing and using data in
6 geographic database.

7 Geographic data are used for various purposes in various different kinds of
8 computing platforms. Geographic data are used in in-vehicle navigation systems to
9 provide drivers of the vehicles with various useful navigation-related and map-related
10 features. Geographic data are also used by navigation-related and map-related services
11 on the Internet.

12 When using geographic data there is often a need to access the data spatially, i.e.,
13 to find data based on the locations of the geographic features represented thereby. One
14 way to access geographic data spatially is to use a spatial index, such as a *k*-d-tree index.
15 A *k*-d-tree index has a tree structure, having a root node, leaf nodes and intermediate
16 nodes between the root node and the leaf nodes. Each node in a *k*-d-tree structure is part
17 of a parent-child relationship. Each parent node includes control information from which
18 one (of two or more) child nodes associated with the parent node can be distinguished
19 based on a search key.

20 When a *k*-d-tree is used as a spatial index, the root node and intermediate nodes
21 include two dimensional spatial boundary information (e.g., latitude and longitude) that
22 relate to the indexed item of data. The indexed item of data can be an item of data that
23 represents a single geographic feature. Alternatively, the indexed item of data can be a
24 collection of data items that represent geographic features encompassed within a bounded
25 area. As an example, a *k*-d-tree can be used as an index to collections (i.e., parcels) of
26 data that represent geographic features located within separate rectangular areas. The
27 nodes of a *k*-d-tree used for this purpose include the boundary information (i.e.,
28 maximum latitude, maximum longitude, minimum latitude, minimum longitude) that

1 define the separate rectangular areas that contain the geographic features represented by
2 the data in the respective parcels.

3 A *k*-d-tree used for organizing geographic data is described in U.S. Pat. No.
4 5,953,722, the entire disclosure of which is incorporated by reference herein.

5 When using geographic data, it is sometimes useful to access the data at different
6 levels of detail (or granularity). Map display and route calculation are examples of two
7 applications in which geographic data are sometimes used at different levels of detail.
8 For example, when using geographic data with a map display application to display a
9 map of an area, a greater level of detail is used for relatively larger scale maps and a
10 lesser level of detail is used for relatively smaller scale maps. Similarly, when
11 performing route calculation, it is often preferable not to consider smaller, low volume
12 roads when determining a route between an origin and a destination.

13 In order to facilitate navigation functions that access and use geographic data at
14 different levels of detail, data that represent geographic features are sometimes organized
15 into layers based on a rank associated with the represented features. The lowest rank
16 (e.g., 0) is associated with those features that are represented only when the finest level of
17 detail is desired. In the case of roads, the lowest rank may be associated with side streets
18 and alleys. On the other hand, the highest rank (e.g., 4) is associated with the most
19 important features, i.e., those that would be displayed even at the coarsest level of detail.
20 In the case of roads, the highest rank may be associated with expressways and major
21 arterial roads.

22 When data representing geographic features are organized into layers, the lowest
23 layer (e.g., 0) includes data representing geographic features of all ranks (e.g., 0-4). A
24 highest layer (e.g., 4) includes data representing geographic features of only the highest
25 rank (e.g., 4). Each other layer includes only those data that represent those geographic
26 features of the associated rank and higher ranks. For example, layer 2 includes data that
27 represent geographic features having ranks 2, 3 and 4. Layer 2 excludes data that
28 represents geographic features of ranks 0 and 1. When data representing geographic
29 features are organized into layers in this manner, a lowest layer has the greatest
30 granularity and the highest layer has the least granularity.

1 These layers can exist as separate collections of the geographic data. When a
 2 navigation function requires the geographic data with a high level of detail, a lower layer
 3 is accessed and used. On the other hand, when a navigation function requires the
 4 geographic data with a low level of detail, a higher layer is accessed and used.

5 When data representing geographic features are organized into separate layers of
 6 granularity, a *k*-d-tree can be used to access the data spatially. A separate *k*-d-tree is used
 7 for each layer.

8 Figure 1 shows a database structure 100 in which the data are organized into
 9 separate layers, labeled 0, 1, and 2. Each separate layer includes a separate collection of
 10 the geographic data, with layer 0 including the data that represent geographic features
 11 ranked 0, 1 and 2, layer 1 including the data that represent the geographic features ranked
 12 1 and 2, and layer 2 including only the data that represent the geographic features ranked
 13 2. Each separate layer also includes its own *k*-d-tree indexing information. Because each
 14 lower layer includes all the items of data represented in all higher layers, many items of
 15 data that represent the same feature are physically stored more than once. As an
 16 example, note that the item of data in the box labeled with an "x" appears in layer 0, layer
 17 1 and layer 2.

18 An index for data can be implemented as either homogeneous or non-
 19 homogeneous. A homogeneous index appends the control information about related
 20 items of data to each indexed item of data. The *k*-d-tree structure in Figure 1 is an
 21 example of an implementation of a homogeneous index. In the embodiment of Figure 1,
 22 appended to each indexed item of data in the database 100 is the control information that
 23 indicates the related items of data. A homogeneous index provides advantages with
 24 relatively small data sets, e.g., those that fit entirely within a given computer's memory.

25 A non-homogeneous spatial index maintains the indexing data in a collection of
 26 small records that are separate from the indexed items of data. A non-homogeneous
 27 index is useful with relatively large data sets, i.e., those that exceed a given computer's
 28 memory. A non-homogeneous index has the benefits of being constructed and
 29 maintained separate from the indexed objects and of requiring relatively little space.

30 A homogeneous index and a non-homogeneous may be used together in a single
 31 database. A single database can include a non-homogeneous index maintained separate

1 from the data being indexed and can also contain homogeneous index information
2 appended to items of data.

3 A *k*-d-tree index (either homogeneous or non-homogeneous) used with a layered
4 database works well for many applications. However, there is room for improvement.
5 For example, it may be desired to provide more granularity, i.e., more layers or ranks of
6 data. Further, it may be desired to make the indexing structure more flexible. Still
7 further, it may be desired to index on additional properties of the data or the represented
8 features.

9 10 SUMMARY OF THE INVENTION

11 To address these and other objectives, the present invention comprises a
12 multi-dimensional spatial index for data that represent geographic features. The
13 multi-dimensional spatial index includes at least two dimensions that define physical
14 boundaries of the geographic features represented by the indexed data. In addition, the
15 multi-dimensional spatial index includes at least one additional dimension that represents
16 another property. The other property is an orthogonal, non-spatial property. The other
17 property may be a property of the geographic features represented by the indexed data or
18 the other property may be a property of the data. The other property represented by the
19 additional dimension may include: granularity, distance, altitude, scale bounds within
20 which each spatial entity is significant, creation date of the data, expiration date of the
21 data or other property. The multi-dimensional spatial index is applicable to the storage,
22 retrieval, visualization and traversal of spatial data sets.

23 24 BRIEF DESCRIPTION OF THE DRAWINGS

25 Figure 1 is a diagram illustrating a conventional *k*-d-tree structure for geographic
26 data.

27 Figure 2 is a diagram illustrating a first embodiment of a database for geographic
28 data with a multi-dimensional index structure having at least three dimensions.

29 Figure 3 is a diagram illustrating a second embodiment of a database for
30 geographic data with a multi-dimensional index structure having at least three
31 dimensions.

1 Figure 4 is a diagram illustrating another embodiment of a database for
2 geographic data with a multi-dimensional index structure having at least three dimensions
3 wherein one of the dimensions is used for layers.

4 Figure 5 is a diagram illustrating an embodiment of a database for geographic
5 data with a multi-dimensional index structure having at least three dimensions wherein
6 one of the dimensions is used for rank.

7 Figures 6A-6D show the different ways that a road segment can be displayed at
8 different scales according to the embodiment in Figure 5.

9 Figures 7A-7D show the different ways that a point of interest or a place can be
10 displayed at different scales according to the embodiment in Figure 5.

11
12 DETAILED DESCRIPTION OF THE
13 PRESENTLY PREFERRED EMBODIMENTS

14 I. FIRST EMBODIMENT – GRANULARITY
15 ADDED AS A DIMENSION TO A *k*-d-TREE INDEX

16 Figure 2 is a diagram illustrating a first embodiment. In Figure 2, data contained
17 in a geographic database 200 are organized using a multi-dimensional *k*-d-tree index
18 having at least three dimensions. Because the multi-dimensional *k*-d-tree index is a
19 spatial index, two of the three dimensions are used to index the data spatially. In the
20 multi-dimensional *k*-d-tree index in Figure 2, the two dimensions used to index the data
21 spatially define latitude and longitude boundaries for the items of data being indexed.

22 In addition to the two dimensions used to index the data spatially, the
23 multi-dimensional *k*-d-tree in the embodiment of Figure 2 includes at least one more
24 dimension. In addition to the two dimensions used to index the data spatially, the
25 multi-dimensional *k*-d-tree in Figure 2 includes a third dimension. The third dimension is
26 used to index another property of the data or of the geographic features being
27 represented. In this embodiment, the third dimension is used as an index for granularity.

28 In the embodiment of Figure 2, the granularity indexing information in the multi-
29 dimensional *k*-d-tree is used to represent three ranks of data. Because rank information is
30 incorporated in the multi-dimensional *k*-d-tree in the embodiment of Figure 2, there is no
31 need to provide separate layers of the data, as in the database shown in Figure 1.

1 Accordingly, a data item which is stored three times in the embodiment of Figure 1 (as
2 shown by the three "x"s) can be stored once in the embodiment shown in Figure 2 (as
3 shown with the one "x").

4 As mentioned above, an index can be implemented homogeneously or non-
5 homogeneously. In the embodiment of Figure 2, the multi-dimensional k -d-tree index is
6 implemented homogeneously, i.e., the indexing information is appended to the data items
7 being indexed.

8 The granularity indexing information used to represent three ranks of data in the
9 multi-dimensional k -d-tree in the embodiment of Figure 2 is not restricted to integer
10 values of rank (i.e., 0, 1, 2, etc.), but may have non-integer values as well (e.g., 0-0.99,
11 1.00-1.99, 2.00-2.99, etc.). Thus, the multi-dimensional k -d-tree index allows a
12 continuous range of rank values, and therefore greater granularity.

13 In the embodiment of the multi-dimensional k -d-tree index in Figure 2, two
14 dimensions are used for spatial organization and one dimension is used for granularity for
15 a total of three dimensions. Alternatively, the data set 200 can be implemented using a
16 six-dimensional k -d-tree. According to this alternative, the six dimensions include four
17 dimensions used for spatial indexing plus two dimensions for the granularity. The four
18 dimensions used for spatial indexing correspond to the boundaries (i.e., minimum
19 latitude, minimum longitude, maximum latitude, and maximum longitude) for each two-
20 dimensional item of data being indexed (i.e., for a rectangular area, its four physical
21 bounds). The two additional dimensions, which are used for the granularity property, can
22 be used to define granularity ranges for each item of data being indexed (e.g., 0-0.99,
23 1.00-1.99, 2.00-2.99, etc.)

24 In the embodiment of Figure 2, the multi-dimensional index includes than three
25 dimensions. In alternative embodiments, the multi-dimensional index may include more
26 than three dimensions. According to alternative embodiments, the multi-dimensional
27 index can include four or more than dimensions. According to these alternative
28 embodiments, two dimensions of the four or more dimensions are used for spatial
29 indexing of the data items being indexed. The other two or more dimensions are used for
30 indexing the items of data with respect to additional properties of the data or the
31 represented geographic features.

1
2 II. SECOND EMBODIMENT – GRANULARITY ADDED
3 AS A DIMENSION TO A NON-HOMOGENEOUS k -d-TREE

4 Figure 3 shows another embodiment of a multi-dimensional k -d-tree index that
5 has at least three dimensions and which is used for indexing data that represent
6 geographic features. In Figure 3, a database 300 for geographic data uses a
7 multi-dimensional k -d-tree index 310. As stated in the Background, an index can be
8 implemented as either homogeneous or non-homogeneous. In the embodiment of
9 Figure 3, the multi-dimensional k -d-tree index 310 is implemented as a non-
10 homogeneous index. Accordingly, the multi-dimensional k -d-tree index 310 is
11 maintained separate from the items of data 320 being indexed.

12 Like the homogeneous multi-dimensional k -d-tree index described in Figure 2, the
13 non-homogeneous multi-dimensional k -d-tree index 310 in Figure 3 is a spatial index.
14 Therefore, two of the three dimensions of the non-homogeneous multi-dimensional k -d-
15 tree index 310 are used for spatially indexing the data 320. In addition to the two
16 dimensions used to index the data spatially, the multi-dimensional k -d-tree index 320 in
17 the embodiment of Figure 3 includes at least one more dimension. In this embodiment,
18 the additional dimension is used as an index for granularity.

19 Like the multi-dimensional k -d-tree index in Figure 2, the multi-dimensional k -d-
20 tree index 310 in Figure 3 can be implemented using six-dimensions, i.e., four
21 dimensions for spatial indexing plus two dimensions for granularity.

22
23 III. ALTERNATIVE EMBODIMENTS

24 A. OTHER PROPERTIES ADDED AS A DIMENSION
25 TO A NON-HOMOGENEOUS k -d-TREE

26 In the embodiments described in Figures 2 and 3, the additional property included
27 as a dimension in the multi-dimensional k -d-tree index was granularity. Granularity is
28 not the only property that can be used as the additional dimension. Instead of granularity,
29 the additional dimension included in the multi-dimensional k -d-tree index can include
30 any other property of the data. Some of the other properties include the following:

- 31 (1) distance,
32 (2) altitude,

- (3) scale,
- (4) rank bounds within which it is contained or significant,
- (5) expiration date, and
- (6) creation date.

"Altitude" as a property refers to the altitude from which a map generated from the data is viewed. For example, for map views representing 10 – 70 meters altitude, certain data representing geographic features are used. For map views representing 71-200 meters altitude, certain other data are used.

"Scale" as a property refers to the map scale used by the map display application that uses the data. At a scale of 1:5000, certain data are used to represent geographic features. At a scale of 1:10,000, other data are used to represent the geographic features.

As items (5) and (6) in the above list indicate, the additional property that is added as a dimension does not have to possess a spatial quality *per se*.

B. ALTITUDE ADDED AS A SPATIAL DIMENSION TO A MULTI-DIMENSIONAL k -d-TREE

In the embodiments described in Figures 2 and 3, two dimensions of the multi-dimensional k -d-tree index are used for spatially indexing the data contained in the geographic database. The two dimensions used for spatial indexing define the latitude and longitude boundaries of the represented features. In an alternative embodiment, the spatial dimensions may also include the altitude of the geographic feature represented by the item of data being indexed. In this alternative, three dimensions of the multi-dimensional k -d-tree index are used for spatially indexing the data contained in the geographic database and an additional dimension, i.e., a fourth dimension, is included for indexing the data based upon another property of the data or the geographic data represented by the data.

According to one of the alternatives described above, a multi-dimensional k -d-tree is disclosed that has at least six dimensions. According to that alternative, two dimensions are used for each spatial direction (i.e., two for latitude and two for longitude) so that four dimensions are used for spatially indexing the geographic data. In that alternative, the multi-dimensional k -d-tree includes at least two additional dimensions

1 which are used for indexing the data based upon another property, e.g., granularity. If
2 altitude is included as a spatial dimension and if two dimensions are used for each spatial
3 direction, an alternative embodiment of a multi-dimensional k -d-tree has at least eight
4 dimensions. In this alternative, six dimensions are used for spatial indexing (two for
5 latitude, two for longitude and two for altitude). In addition to these six dimensions, the
6 multi-dimensional k -d-tree includes at least two additional dimensions. These additional
7 dimensions are used for indexing the data based upon a property of the data other than
8 latitude, longitude or altitude of the represented feature, e.g., granularity.

10 C. OTHER ALTERNATIVES

11 As mentioned above, the multi-dimensional k -d-tree index can be implemented as
12 either homogeneous (as in Figure 2) or non-homogeneous (as in Figure 3). The multi-
13 dimensional k -d-tree index may be implemented as a homogeneous index for small,
14 malleable data sets. The multi-dimensional k -d-tree index may be implemented as non-
15 homogeneous for larger or constant data sets stored on magnetic media or CD-ROM.

16 In the previously described embodiments, the multi-dimensional index is
17 implemented using a k -d-tree structure. In alternative embodiments, the multi-
18 dimensional index may have a structure other than a k -d-tree. As an example, for the
19 multi-dimensional index may have a different hierarchical data structure, such as those
20 from the field of solid modeling, as described by Hanan Samet in *The Design and*
21 *Analysis of Spatial Data Structures*, Addison-Wesley, Reading, MA, 1990, Chapter 5.

22 In the embodiments described above, one additional property is indexed in the
23 multi-dimensional k -d-tree in addition to the spatial dimensions. In alternative
24 embodiments, the multi-dimensional k -d-tree can include dimensions for more than one
25 additional property. For example, a four dimensional k -d-tree can be used for indexing
26 latitude, longitude, granularity and expiration date of the represented data items.

28 IV. APPLICATIONS

29 This section presents examples to illustrate applications in which the advantages
30 of the disclosed embodiments can be realized.

1
2 A. Map Storage

3 The embodiments of the multi-dimensional k -d-tree index, described above, allow
4 infinite granularity within a single spatial data file (subject to the number of bits of
5 resolution allocated to representing the granularity). Only a small percentage of entities
6 in a data set are relevant at the highest layer, however, every entity in a spatial data set is
7 relevant at the lowest layer.

8 Figure 4 depicts a database 400 containing geographic data 420. The geographic
9 data 420 includes three layers of granularity formed by three layers 422, 424, and 426
10 stored in a single file. A single non-homogeneous multi-dimensional k -d-tree spatial
11 index 410 is used to access the data 420. If a conventional k -d-tree index were used, the
12 same spatial data set would have required three two-dimensional spatial indices and
13 stored in three progressively larger data files – each duplicating the higher layers' data.

14 Although three layers 422, 424, and 426 are shown in the embodiment of
15 Figure 4, alternative embodiments may include more or fewer layers. In addition, the
16 layers are not constrained to integer values but may assume a continuous range of values.

17
18 B. Map Rendering

19 The embodiments of the multi-dimensional k -d-tree index, described above, allow
20 precise control over the range of scales at which spatial entities are visualized. Each
21 cartographic feature can possess a specific range of scales at which it is to be visualized.
22 A minor feature might, because of its prominence or location, be visualized at
23 uncharacteristically large map scales. By contrast, with conventional indexing schemes,
24 cartographic features are visualized depending on broad, general criteria.

25
26 C. Navigation routing

27 The embodiments of the multi-dimensional k -d-tree index, described above, allow
28 for several rank-dependent representations of a navigable road segment, not just in a
29 single spatial index but in a single navigable data set as well. High-ranking roads, for
30 example, might be aggregated into progressively simpler representations as lower-

1 ranking intersections are dropped. At a highest rank, a stretch of highway might be
2 represented as a single line between two major junctions. At a lowest rank, the same
3 stretch of highway might be represented as a string of shorter segments connecting the
4 intersections that lie between the two major junctions.

5 Figure 5 depicts a geographic database 500 including data 520 stored with three
6 ranks of granularity, 522, 524, and 526. A single non-homogeneous, multi-dimensional
7 *k*-d-tree index 520 is used to access the data 520. (Although three ranks are shown, the
8 multi-dimensional *k*-d-tree index allows a continuous range of rank values.) By contrast,
9 with prior indexing schemes, the same data would have required three two-dimensional
10 spatial indices and three somewhat redundant, progressively smaller rank files.

11 12 V. REPRESENTATION OF SPATIAL ENTITIES

13 Embodiments of the multi-dimensional *k*-d-tree index, described above, allow for
14 several scale-dependent representations of an entity within a single spatial index. A
15 divided highway, for example, might have a different representation for each of the map
16 scales at which the data can be viewed. Figures 6A-6D depict four representations of a
17 divided highway viewed at different map scales. (Although four map scales are
18 suggested, embodiments of the multi-dimensional *k*-d-tree index allow for a continuous
19 range of map scale values.) With prior indexing schemes, a separate two-dimensional
20 index (and possibly a separate data set) would be required to distinguish each of the four
21 representations.

22 Embodiments of the multi-dimensional *k*-d-tree index, described above, facilitate
23 distance-dependent representations of a three-dimensional entity within a single spatial
24 index. A sphere, for example, might be represented by tessellated polygons of increasing
25 complexity depending on the distances from which it is viewed. Figures 7A-7D depict
26 representations of varying complexity for a single three-dimensional sphere viewed from
27 various distances.

VII. ADVANTAGES

Several advantages follow from embodiments of the disclosed systems.

A. Allows continuous rather than stepped granularity.

Prior indexing techniques organized spatial data into stepped planes of decreasing granularity (e.g., layers 0, 1, 2 . . . n). The disclosed embodiments index spatial data according to their granularity as well as their physical dimensions, and therefore the granularity values are allowed to fall more precisely within a continuous range rather than a series of steps. For example, a road segment can have a rank of 1.375. A graphic of the road segment can be visualized at map scales up to exactly 1:9,056. A navigable road segment, on the other hand, can fall within a range of (possibly fractional) rank values.

B. Eliminates the need to parcelize spatial data into separate but coincident "layer" files.

Conventional indexing techniques require a spatial data set to be distributed among a series of "layer" files, each containing the data relevant to a certain degree of granularity (less granular data is often replicated among several, possibly all, layers). Each layer file then indexes its spatial entities using a traditional two-dimensional spatial index according to their minimum bounding rectangles: \min_x , \min_y , \max_x , and \max_y . The third dimension can further index spatial data by \min_z and \max_z values indicating their bounding layers, eliminating the need to parcel a spatial data set into layers of various granularities.

C. Allows a single spatial data set to be visualized at any granularity.

Visual displays of two-dimensional spatial data face the same granularity issues as the physical storage format. Certain types of spatial entities (points of interest, local streets) lose their significance at higher map scales. Linear or polygonal graphics shrink to invisibility at higher map scales. The third dimension afforded by the Invention can

1 further index spatial data by \min_z and \max_z values indicating the minimum and
2 maximum altitudes or map scales within which each spatial entity should be visualized.

3
4 D. Allows multiple scale-dependent representations
5 of a spatial entity in a single collection of graphics.

6 Visual displays of two-dimensional spatial data often use one of several different
7 graphics to represent a spatial entity, depending on the scale or altitude at which the data
8 is visualized. The third dimension afforded by the multi-dimensional k -d-tree index can
9 index the minimum and maximum altitudes or map scales at which to display each
10 graphic representation, allowing a single collection of graphics to distinguish among
11 several scale-dependent representations of a spatial entity.

12
13 E. Eliminates the need to parcel navigable data into
14 separate but coincident "rank" files, and allows multiple
15 rank-dependent representations of a navigable entity.

16 Navigation databases face the same granularity issue as physical storage formats
17 and visualization. Like the "layer" sets into which a spatial data set is often parceled,
18 navigation data sets are parceled into separate "rank" sets, each containing the data
19 significant at a certain granularity or level of functionality. Like graphics collections,
20 each navigation data set may use multiple rank-dependent representations of a navigable
21 entity. At the lowest rank (most granular), an arterial road might consist of a number of
22 connected segments joined at side-street intersections. At higher ranks (less granular),
23 the arterial road might be aggregated into a single segment connecting other arterial
24 roads. At the highest rank (least granular), the arterial road might not even be
25 represented. Embodiments of the disclosed multi-dimensional index includes a third
26 dimension that facilitates indexing navigable entities by minimum and maximum rank,
27 allowing a single navigable data set to distinguish among several rank-dependent
28 representations of each navigable entity. Embodiments of the disclosed multi-
29 dimensional index also eliminate duplication of entities (points of interest, higher-rank
30 intersections) that are shared among ranks.

- 1 It is intended that the foregoing detailed description be regarded as illustrative
- 2 rather than limiting and that it is understood that the following claims including all
- 3 equivalents are intended to define the scope of the invention.

009077 22090260
00706927 440600